Remarks

This Response and Amendment is being filed in response to the final office action dated April 6, 2007 in the referenced application. The Applicant notes with appreciation the Examiner's efforts in advising the Applicant regarding the language of the claims to resolve the rejections under § 101. It is the Applicant's understanding the a tentative approval of the language of the independent claims of the application has been provided subject to the review of the Examiner's Primary Examiner.

The Examiner has rejected claims 1, 3, 5-7, 11, 13, 15-35, 40-46, 49, 52-53, 60-62 and 82-96 under 35 U.S.C. § 101, stating that the claims of the invention are directed to non-statutory subject matter. The Applicant wishes to state for the record that they are still in disagreement with the interpretation of the Examiner and the Primary Examiner of the sections of the MPEP regarding computer related non-statutory subject matter (§ 2106.01) and their assertion that the lack of a recitation of functionality within the claims renders the claims (directed to a data structure) as per se non-functional descriptive material. According to the Applicant's reading of the relevant sections of the MPEP and the applicable case law, the recording of ***functional*** descriptive material (a data structure is considered to be in this class) on a computer medium makes it structurally and functionally inter-related to the computer software and hardware, and thus statutory. *See MPEP § 2106.01.* That is, the recording of a data structure on a computer medium allows an interaction between the computer and the data structure which provides the normal functionality that one would expect of a data structure, that is the manipulation of the data stored in the data structure by the computer. As a result, the Applicants respectfully submit that there is no need to add functional limitations to the claims.

Note that on page five of the office action, the Examiner includes an excerpt from the MPEP § 2106.01 with particular text in bold and italics which states as follows: "When nonfunctional descriptive material is recorded on some computer-readable medium, in a computer or on an electro-magnetic carrier signal, *it is not statutory since **no requisite functionality** is present to satisfy the practical application requirement*." The Applicant notes that this particular restriction applies to only non-functional descriptive material. The Applicant strongly disagrees that the data structure specified in the independent claims of the application represents non-functional descriptive material, even absent a recitation of functionality in the claims. The MPEP is clear in stating that data structures represent functional descriptive material which becomes statutory when recorded on a computer-readable medium, thereby imparting the normal functionality which one expects from a data structure, that is the manipulation of data.

However, in an effort to advance the prosecution of this case, the Applicant has agreed to insert a recitation of functionality within the independent claims. With respect to Claim 1, the Applicant has inserted the limitation in the claims that the arrangement of and information stored in the fundamental data structures allows the addition, removal and searching of data items stored in the data structures. The collection of data structures and their inter-relations forms a data management system, which would normally allow these types of operations with respect to the data stored therein. A similar type of limitation has been added to claim 85 of the application, which is similar to claim 1.

The other independent claim of the application, claim 52, also claims a data management system, however, this particular claim already has a recitation of functionality therein, containing the limitation that the encapsulated references to associated data structures are also logical indices which uniquely identify the encapsulated data instance and also encode the location of each of the data instances on a computer readable media. Therefore, the function of locating this data on a computer readable media is recited in this claim, thereby rendering this claim statutory.

Because the independent claims of the application are now believed to be statutory, the Applicant respectfully submits that all dependent claims thereof are statutory as well, and requests that the Examiner's rejection in paragraph 5 of the office action under 35 U.S.C. § 101 be withdrawn.

The Examiner has rejected claims 1, 3, 7, 9, 11, 13, 15, 16, 53, 82-87, 89 and 92 under 35 U.S.C. § 103(a) as being unpatentable over White in view of Abineri. As stated in each of the preceding responses to office action, the Applicant respectfully submits that the application of White is not proper in this instance because White teaches away from the invention in requiring that both its data items and its associated type and relationship information be stored in relational database tables. This is clearly evident in Figures 2 and 3 of White which show this information being stored in tables, and in various other places in the White specification, in which a description of the use of tables in White is explained. These passages of White will not be repeated here, as they have been quoted in previous responses and are part of the record of the case.

As a result of this teaching of White, the Applicant submits that the combination of White with any other reference is improper because White unequivocally teaches away from the present invention. The Applicant has amended the independent claims of the application to recite the limitation that the data be stored in non-tabular form, which further clarifies the distinction between White and the present invention, which specifically teaches away from the use of tables of any kind.

The Examiner states that White does not explicitly teach the limitation "in non-tabular form." The Applicant respectfully submits that White not only does not teach this limitation but explicitly teaches away from this limitation. The Examiner further states that Abineri teaches this limitation in paragraphs 61-66. Abineri teaches a display generator for an object oriented database. Paragraphs 61-66 describe a tree structure consisting of objects having a parent/child relationships with other objects. The Applicant directs the Examiner's attention to paragraph 35 of Abineri, which describes the tree structure and the objects and classes therein. Paragraph 40 describes the primary objective of Abineri, which is the conversion of a flat database file (i.e., a non-object oriented file) to an object orientated database for purposes of display in a tree structure. Abineri never actually discusses the structure of the original flat file database nor does he discuss the implementation of the object oriented database, which can be implemented in any manner well known in the art, including in a tabular form as described in White. Because there is no specific teaching of a database in non-tabular form in Abineri, the Applicant respectfully submits that this limitation of the claims is not taught by the combination of White and Abineri.

Even for the sake of argument, if Abineri did disclose a database in non-tabular form, it makes no sense to combine the teaching of a database in non-tabular form with a database that ***requires*** a tabular form, as does the White database. As a result, the Applicant respectfully submits that this combination does not teach what the Examiner states and further that the combination of the two references is improper due to a lack of teaching, suggestion or motivation within either of the references to make the combination.

To further make the distinction between the data structures of the present invention and the relational tables of White, claim 1 has been modified to remove the phrase "common fundamental data structures" and to replace it with the phrase "independent data structures having a common form" to further indicate that the structures in which individual data objects are stored are in no way structurally related to each other, except through encapsulated references which indicate some type of association. This removes any and all confusion that the word "common" might mean "shared" or "stored together" in the claims.

With respect to claim 3, neither White nor Abineri shows the encapsulating of the references to data structures containing associated data instances within a data structure containing the data instances with which they are associated. In White, associations between data objects are kept in a relational database table, along with references to the data objects themselves, which are also stored in a separate relational database tables. Abineri does not teach a specific structure. Therefore, the combination does not teach storing data objects in a data structure along with references of associated data objects.

Claim 7 contains the limitation that the encapsulated references are at least one dimension and that the dimensions correspond to a type of association. This is also not taught in either White or Abineri. In Abineri, the associations are simply "contained in" or "containing" type relationships (i.e., typical object-oriented relationship between classes and sub-classes) which do not carry a context of the type of associations. In White, the types of associations are stored in a relational database table which shows the relationship between two data objects. In the present invention, the types of associations are indicated by how the references are grouped together. All encapsulated references to associated data instances which are grouped together have the same type of association with the encapsulating data instance as the others in the group. Different groups denote different types of relationships with the encapsulating data instance.

With respect to claims 9, 11, 13, 15-16 and 53, which state that the data structures are application- independent, this is generally true of all databases which can be acted upon by many different applications. These claims are patentable by virtue of their dependence from other claims which the Applicant submits are patentable.

With respect to claims 82-87, the Applicant respectfully submits that neither White nor Abineri teaches a system in which all associated data instances may be encapsulated within a single data instance. Instead, the associations in Abineri happen through a tree structure in which each of the objects in the tree may have only one parent but multiple children. *See* Figures 3a and 3b for a pictorial representation of the tree structure. This limitation is not present in the present invention, which instead allows all associated data instances to be encapsulated within each individual data instance. Claims 83 and 84 are dependent upon claim 82.

Claim 85 contains limitations very similar to those of claim 1 and for the same reasons as respect to claim 1, the combination of White and Abineri do not render this claim obvious and, as with respect to claim 1, the applicants respectfully submit that the combination of White and Abineri is improper as lacking any teaching, suggestion or motivation to make the combination. Further, White and Abineri teach away the present invention and from each other in that White requires tables to be used and the Examiner claims that the Abineri does not require tables to used (i.e., "non-tabular form"), although the Applicant questions that conclusion with respect to Abineri.

Claim 89 claims that associated items are arranged in sets within each encapsulation which define the types of associations between the items. This is similar to claim 7 and the same arguments apply here as well. That is, that associated items are arranged in sets which define the type of association between them and the encapsulating data instances. Neither White nor Abineri nor the combination thereof discloses this limitation. With respect to claims 7 and 89, the Examiner cites White at column 7, line 5-11. This passage of White clearly refers not to the types of associations between objects but to the types of the objects themselves. The Applicant directs the Examiner's attention to the cited White passage wherein it states the following: "As shown in FIG. 3, a Type Table Entry for a given object type includes one or more object identifiers (or pointers or keys) that identify the objects that belong to a given object type." What is discussed here is a entry in a table for a given object type and a list of objects having that type. No mention is made in this passage of the ***type of association*** between objects but merely the type of the objects themselves.

With respect to claim 89, the Examiner sites column 7, lines 44-61 of White as disclosing the concept of claim 89, which is that encapsulated references are grouped in sets defining the type of their relationship with the encapsulating reference. This section of White, however, refers to a relational database table entry for a given type of relationship which is referred to by another table entry which contains pointers to the related data objects. This is fundamentally different from the present application, in which a set of identifiers are first stored within an encapsulation of the data item to which they are related and secondly, are grouped within that encapsulation in sets which define the relationship between that set of data items and the data item which is encapsulating the set of pointers. Note that no tables are used in implementation of the present invention.

The Examiner has rejected claims 5, 8, 18-24, 31-34, 36, 47-48, 50-52, 54-60, 62, 88, 90 and 93 as being unpatentable over White in view of Abineri and further in view of Koenke, et al. The Applicant's previous comments with respect to White and Abineri apply here as well, that is, the combination of these two references is improper because White teaches away from the teachings of the present invention and, in fact, teaches away from Abineri in that it requires the use of tables, while the present application cannot work with tables. The Examiner states that the combination of White and Abineri fails to teach the limitation that the logical index is m dimensional and has n bits per dimension, but that Kroenke teaches this limitation. First, the Applicant respectfully submits that Kroenke, which teaches a computer system for allowing a user to create a relational database scheme, also teaches away from the present application. Kroenke refers to a model which uses a relational database table, while the present application specifically rejects that particular type of database in favor of the data instance centric model presented in the claims (i.e., each data instance contained in an independent data structure of common form). Further, the Examiner states that Kroenke teaches an object data model with indices which are m dimensional and have n bits per dimension. However this is not disclosed in the cited passage in Kroenke, which describes a subscript for an attribute consisting of a pair integers having the form "m.n", wherein "m" refers to the minimum cardinality and "n" refers to the maximum carnality of the attribute. The minimum cardinality is a minimum number of instances of the attribute in the data record and the maximum cardinality is a maximum number of instances of the attribute in the data record. The Examiner's interpretation of Kroenke appears to be incorrect in that it does not disclose an "m dimensional" index but discloses a one dimensional attribute having a minimum number of instances and a maximum number of instances. This is not the same as the "m dimensional" index referred to in claim 5. Therefore, for this additional reason, the Applicant respectfully submits that the combination of White, Abineri and Kroenke does not disclose the claimed subject matter and further that Kroenke teaches away from the present application which makes it application to and combination with the other references improper as having no teaching, suggestion or motivation to do so.

With respect to claim 8, the Examiner states that White in view of Kroenke teaches the limitation of at least one dimension having a plurality of encapsulated references, citing White at column 7, lines 5-11 and column 7, lines 45-52. As previously stated, White in column 7 at lines 5-11 discloses a relational database table which has table entries defining the types of data objects and at column 7, lines 45-52 a relational database table having table entries for defining the types of relationships between data objects. As previously stated, the use of the relational database tables in White teach away from the present application, which encapsulates all relationship information within an independent data structure for each data object. As a result, these portions of White are not applicable to the present application. The Examiner further states that this concept is disclosed in Kroenke at column 6, lines 26-65. This portion of Kroenke discloses an object orientated system and describes, as is shown in Figure 2, the value attributes and group attributes and the types of the attributes which are found within a particular semantic representation of an object. Nothing in this portion of Kroenke or anywhere in Kroenke is taught an multi-dimensional index wherein each dimension of the index is related to a plurality of encapsulated references. Kroenke, in the cited passages, merely describes a typical object oriented class structure wherein attributes are assigned to objects and inherited by instances of the objects.

Claims 18-22 refer to operations on the data management system of the present application which may be implemented in any database management system such as adding, removing and searching data objects, and as such are patentable based on their dependence from other claims.

Claim 23 differs from a normal data management system in that the selection criteria used when searching is itself an encapsulated data instance present in the database. Note that claim 23 states "accessing references encapsulated with said known encapsulated data instances representing said selection criteria". This places the selection criteria as an item or items within the database management system and which would encapsulated references to all other data instances which meet the selection criteria. This concept is unknown in any database management system of which the Applicant is aware and certainly not disclosed in Kroenke. The Examiner states that this particular attribute is disclosed in Kroenke in column 12, lines 15-44. This passage describes object link profiles which are used to relate one semantic object to another within the semantic object data model. However, it does not describe searching using selection criteria stored as an item or items in the data base.

With respect to claim 24, the Examiner states that White in view of Kroenke teaches the limitation that each of the dimensions of the logical index corresponds to a type of association. The Applicant's remarks with respect to claim 7 apply here as well. Neither White nor Kroenke disclose a database management system in which the grouping of encapsulated references corresponds to the type of association present between the grouped references and the encapsulating data instance.

With respect to claim 36, this claim contains the limitation that the encapsulated references are a logical index which uniquely identifies the data instance and also encodes the physical location of the associated data instances on the computer readable media. That is, the unique reference to each data instance not only provides a unique reference but also provides the physical location of the record on the computer readable media, whether it be a hard disk or a random access memory of the computer. Neither White nor Kroenke teach this limitation. The cited portions of White in columns 6 and 7 teach table entries having pointers to entries in other tables, such as might contain relationship information. No where in Kroenke or White is it disclosed that each data object has a unique reference nor does it disclose a unique reference also encoding a physical location on a computer readable media. This is simply not disclosed in any of the references cited.

With respect to claim 23, the Examiner states that White teaches the limitation that an identity in one or more of the m dimensions in the logical index indicates a member item in a container. Once again the Examiner refers back to the portion of table 7 which refers to data object types within a relational table. White does not teach a multi-dimensional index and, therefore, cannot teach that an identity in one of the dimensions indicates membership in a container item. White, in fact, does not appear to discuss container items. The cited portion of White discloses a table containing object data types. This portion of White does not refer to membership in a container or, for that matter, any type of relation between any items in the database.

Claims 27-30 have been rejected under 35 U.S.C. § 103(a) as being unpatentable over White in view of Abineri and further in view of Kroenke and further in view of Walker. The Examiner refers to Walker to teach the limitations of Boolean mathematical operations, which the Applicant is willing to stipulate are well known in the prior art. However, within the context of the database system presented in the parent claims of claims 27-30, the use of Boolean mathematical operations to sort through lists of encapsulated data instances for purposes of satisfying search criteria is not disclosed in the cited references.

The Examiner has rejected claim 40 under 35 U.S.C. § 103(a) as being unpatentable over White in view of Abineri and further in view of Bielak, stating that the combination of White and Abineri does not teach a plurality of encapsulated references representing ASCII characters, but that this is taught in Bielak. In the scope of the present application, encapsulated data instances representing ASCII characters provides for the inclusion of a single ASCII character as a single database item to which other database items are able to refer by encapsulation. This portion of Bielak merely teaches that data may be stored in ASCII format. The use of the term "ASCII" in this format does not indicate that the claimed limitations are disclosed in that reference. Bielak does not come close to teaching that individual ASCII characters may be stored as individual data items in a database and be referred to by other data items which require the use of that particular ASCII character. The same argument applies for claim 42 regarding Unicode characters for which the Examiner has cited Eversol, and claim 44 wherein data instances representing the token members of the random token set of any data type for which the Examiner has cited Schwartz.

The Examiner has rejected claims 17, 49 and 61 under 25 U.S.C. § 103(a) as being unpatentable over White in view of Abineri and further in view of Silberberg, et al. stating that Silberberg teaches the limitation of encapsulated references being a reference to data instances in another competing environment. Silberberg does teach distributed database accessing wherein other databases could be considered "other computing environments." However, Silberberg teaches a specific architecture for accessing the information stored in other computer environments which is different than the architecture described in the present application. The limitation of the parent claims to claim 17, 49 and 61 describe this architecture and are not disclosed by the combination of White and Abineri as discussed in detail above.

The Examiner has rejected claims 94-96 under 35 U.S.C. § 103(a) as being unpatentable over White, in view of Abineri and further in view of Suver, stating that Suver discloses, at column 10, lines 9-27 the encapsulation of embedded elements. First, Suver teaches a system for embedding information in object-relational databases. The Applicant points out that the present application describes neither an object oriented nor a relational database but instead describes a "data instance centric" database. Object orientated databases describe databases in which attributes are inherited from parent to child depending on the type of data being represented in the data object. Relational databases are databases set up using relational database tables. Therefore, the Applicant respectfully submits that there is no teaching, suggestion or motivation for citing Suver with respect to the present application. Nevertheless, Suver does not teach the embedding of objects within encapsulating data objects but includes the embedding of elements within records in a relational data base table. Therefore, the Applicant respectfully submits that Suver has been misinterpreted by the Examiner as disclosing embedding of objects within an encapsulated data instance.

Conclusion

The Applicant has provided reasoned arguments regarding the § 101 rejection and has provided amendments to the claims which should overcome those rejections.

In addition, the Applicant has pointed out that the combination of White and Abineri, either alone or in combination with any other reference is improper with respect to the present application because White teaches away from the present application in that it requires tables and White and Abineri appear to teach away from each other in that White requires tables and Abineri (according to the Examiner) appears to disclose a system which does not require tables. Therefore, there is no teaching, suggestion or motivation to combine these two references to make the rejections made by the Examiner. These comments apply to all rejections based upon the combination of White and Abineri or upon merely the application of White to the present application.

The Examiner has stated in a previous office action that White is not limited to one embodiment but teaches features for embedding relationship information within data objects. The Applicant respectfully submits that any interpretation of White not requiring the storage of objects, object types and object relations in relational database tables is improper as that is the only embodiment taught by White. White does teach features for embedding data within data objects but there is no teaching in White of the use of this particular feature for encapsulating the relationship between data instances, which are handled in the manner already described. The interpretation of White in the manner described by the Examiner leads to the question of why there would be two ways of doing the same thing in White (i.e., using relational tables to indicate associations and also using embedded data to point to associated data objects). This simply does not make sense as there is no explanation in White for when one method would be used in favor of another, and, in fact, no explanation of the use of embedded data for this purpose at all. The Examiner's interpretation of White in this manner cannot stand. In addition, White is devoid of any disclosure for grouping sets of encapsulated references to indicate a particular type of relationship between the set and the encapsulating data instance. The interpretation of White in any other manner is improper and is hindsight construction of that reference on the part of the Examiner. White very explicitly teaches in the opposite direction from the present application in that tables are required for the storage of the various types of information.

The Applicant therefore respectfully requests that the Examiner withdraw the 101 rejections and all other 103 rejections based upon any application of the White reference.

The Examiner has indicated that various claims of the application are allowable if rewritten in independent form but the Applicant declines to do so at this time pending the outcome of the prosecution of the other claims of this application.

The Applicant wishes to thank the Examiner for his time in review of proposed new claim language to traverse the § 101 rejection and the Examiner's consultation with the Primary Examiner in this regard. Should the Examiner have any questions, the Applicant refers the Examiner to the Applicant's attorney whose contact information is listed below.

Respectfully submitted,

Dennis M. Carleton
Reg. No. 40,938
FOX ROTHSCHILD LLP
625 Liberty Avenue, 29th Floor
Pittsburgh, PA 15222
(412) 394-5568